*Bioinformatics*, 33(4), 2017, 580–582
doi: 10.1093/bioinformatics/btw689
Advance Access Publication Date: 29 November 2016
Applications Note

Sequence analysis

# InMoDe: tools for learning and visualizing intra-motif dependencies of DNA binding sites

**Ralf Eggeling[1,*], Ivo Grosse[2,3] and Jan Grau[2]**

[1]Helsinki Institute for Information Technology (HIIT), Department of Computer Science, University of Helsinki, Helsinki, Finland, [2]Institute of Computer Science, Martin Luther University Halle–Wittenberg, Halle (Saale), Germany and [3]German Centre for Integrative Biodiversity Research (iDiv) Halle-Jena-Leipzig, Leipzig, Germany

*To whom correspondence should be addressed.

Associate Editor: John Hancock

Received on August 26, 2016; revised on October 20, 2016; editorial decision on October 25, 2016; accepted on October 27, 2016

## Abstract

**Summary:** Recent studies have shown that the traditional position weight matrix model is often insufficient for modeling transcription factor binding sites, as intra-motif dependencies play a significant role for an accurate description of binding motifs. Here, we present the Java application InMoDe, a collection of tools for learning, leveraging and visualizing such dependencies of putative higher order. The distinguishing feature of InMoDe is a robust model selection from a class of parsimonious models, taking into account dependencies only if justified by the data while choosing for simplicity otherwise.

**Availability and Implementation:** InMoDe is implemented in Java and is available as command line application, as application with a graphical user-interface, and as an integration into Galaxy on the project website at http://www.jstacs.de/index.php/InMoDe.

**Contact:** ralf.eggeling@cs.helsinki.fi

## 1 Introduction

The position weight matrix (PWM) model (Stormo *et al.*, 1982) has been the standard choice for representing the statistical properties of functional DNA elements such as transcription factor binding sites for more than three decades. While easy to interpret, use and visualize, the original PWM model makes the assumption that all nucleotides contribute independently to the total binding affinity. Recent studies have shown that taking into account dependencies among nucleotides can yield a better motif representation (Eggeling *et al.*, 2014a; Keilwagen and Grau, 2015; Siebert and Söding, 2016; Zhao *et al.*, 2012), and that the DNA shape (Mathelier *et al.*, 2016) could serve as one biophysical explanation for deviations from the independence assumption.

In this article, we present InMoDe, a user-friendly suite of multiple tools for learning **in**tra-**mo**tif **de**pendencies in various scenarios. InMoDe is based on *parsimonious context trees* (PCTs) as proposed by Bourguignon and Robelin (2004), which provide a sparse parameterization of the conditional probability distribution for avoiding overfitting. The position-specific use of PCTs yields an *inhomogeneous parsimonious Markov model* (iPMM). Both model structure and parameters of an iPMM can be robustly learned without resorting to computationally expensive parameter tuning even when latent variables are involved (Eggeling *et al.*, 2015). While finding optimal PCTs is computationally hard, recent algorithmic advances allow us to solve typical instances fast enough to effectively consider dependencies up to order six (Eggeling and Koivisto, 2016). InMoDe also provides tools for applying learned models, such as scanning given sequences for statistically significant hits and classifying binding sites. The learned dependencies of an iPMM can be directly visualized by a conditional sequence logo (CSL, Fig. 1A), which can be viewed as PCT-based extension of a traditional sequence logo (Schneider and Stephens, 1990). InMoDe is implemented in Java using the Jstacs library (Grau *et al.*, 2012) and provides three user interfaces: a command line interface, a GUI and an integration into Galaxy workflows. In the next two sections, we briefly discuss the different components of InMoDe, consisting of four learning modes and three application tools. For a detailed documentation of all features we refer to the user guide on the project website.

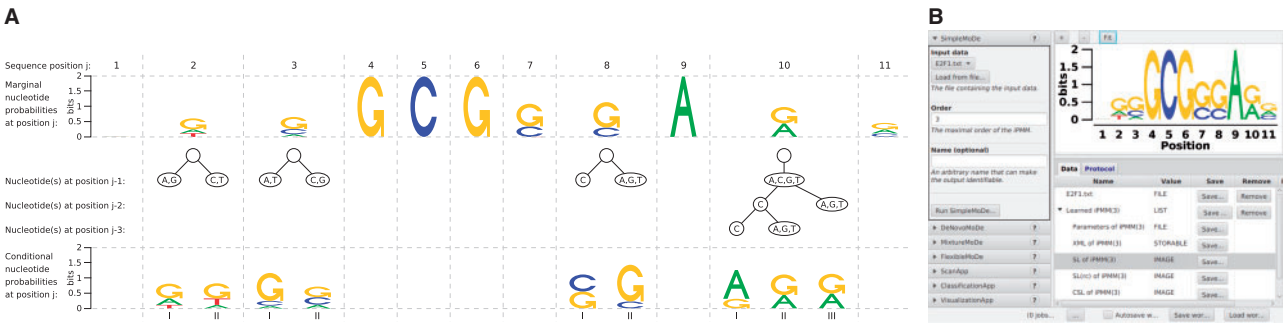


**Fig. 1.** Exemplary conditional sequence logo as generated by `VisualisationApp` (**A**) and screenshot of the graphical user interface (**B**). The examples use the E2F1 dataset from JASPAR (Sandelin *et al.*, 2004). The top row of nucleotide stacks in the CSL corresponds to a classical sequence logo (displayed in top-right box of the GUI), whereas the bottom row shows conditional nucleotide probabilities given the context represented by the PCT (middle). A motif refinement by first-order dependencies is achieved at position 2, 3 and 8, higher-order dependencies do occur once at position 10, and yet the model allows to choose for simplicity where nothing else is justified by the data, such as at position 4–7 and 11

## 2 Learning modes

In the most simple scenario, the input data is a set of gapless aligned binding sites of identical length, which might have been either obtained experimentally or from a database. Structure and parameters of a single iPMM can be learned exactly using the tool `SimpleMoDe`, which implements the methodology of Eggeling *et al.* (2014b).

In the case of, e.g. ChIP-seq positive fragments, the input sequences are of arbitrary length and the location of the binding sites is not known, leading to the problem of *de novo motif discovery*. Here, learning is computationally more expensive as it involves latent variables that indicate motif occurrence and strand orientation of a putative binding site in each sequence. The tool `DeNovoMoDe` implements a recent algorithm for learning iPMM s in this scenario (Eggeling *et al.*, 2015), which is a variant of a stochastic EM algorithm (Nielsen, 2000), but uses the BIC-score (Schwarz, 1978) of the optimal iPMM as target function.

In a third scenario, we again assume the input sequences to be pre-aligned and of same length. But in contrast to the first scenario, not one single, but a mixture of $K$ component models is to be learned. Here, latent variables are involved as well and indicate the mixture component. A possible use case is to investigate to which degree complex intra-motif dependencies could be explained by two (or more) different models of lower complexity. The tool `MixtureMoDe` allows this type of analysis, using essentially the same algorithm as `DeNovoMoDe` with the sum of the $K$ BIC-scores as target function.

Finally, the tool `FlexibleMoDe` combines the functionality of the three previous tools, by taking into account latent variables for motif position, strand orientation and motif type. While it contains the other three tools as special cases, it allows a wide range of additional scenarios for less common use-cases. In addition, `FlexibleMoDe` allows for learning from weighted data, which can be useful if sequences should contribute to a different degree, e.g. according to ChIP-scores, to the learned model.

## 3 Application tools

ScanApp scans a given dataset for high-scoring occurrences of a previously learned iPMM. In order to assess which likelihood value is sufficient for declaring a hit, it determines a cutoff threshold based on a user-specified false positive rate (FPR) for occurrences on a negative dataset. A negative dataset can be either given as user input or it is constructed as a randomized version of the positive dataset.

`ClassificationApp` performs a binary classification of sequences, where the classes are represented by an iPMM each, thus it accepts only sequences of the same length as the models. If no background iPMM is available, either a pseudo-background based on a homogeneous Markov chain originating from a user-specified negative dataset or a PWM model with uniform parameters can be used instead.

All learning tools automatically produce a standard visualization of structure and parameters of the learned iPMM(s) in terms of a CSL. The tool `VisualisationApp` allows to plot customized CSLs in order to obtain an appropriate degree of detail according to publication format and target audience. For example, Figure 1 displays a default CSL, which additionally contains descriptions of the main plot elements on the left.

## 4 Conclusions

InMoDe provides multiple tools for learning, utilizing and visualization intra-motif dependencies within functional DNA elements. While the main use case is modeling of transcription factor binding sites, other types of functional nucleotide elements that may be represented by sequence motifs can be analyzed as well.

## Funding

This work was supported in part by the Academy of Finland, Grant 276864, and by a grant from the Deutsche Forschungsgemeinschaft (GR 4587/1-1).

*Conflict of Interest*: none declared.

## References

Bourguignon,P.Y. and Robelin,D. (2004). Modèles de Markov parcimonieux: sélection de modele et estimation. In: *Proceedings of JOBIM*.

Eggeling,R. and Koivisto,M. (2016). Pruning rules for learning parsimonious context trees. In: *Proceedings of UAI*, vol. 32, pp. 152–161. AUAI Press.

Eggeling,R. *et al.* (2014a) On the value of intra-motif dependencies of human insulator protein CTCF. *PLoS One*, **9**, e85629.

Eggeling,R. *et al.* (2014b). Robust learning of inhomogeneous PMMs. In: *Proceedings of AISTATS*, pp. 229–237.

Eggeling,R. *et al.* (2015) Inferring intra-motif dependencies of DNA binding sites from ChIP-seq data. *BMC Bioinform.*, **16**, 375.

Grau,J. *et al.* (2012) Jstacs: a Java framework for statistical analysis and classification of biological sequences. *J. Mach. Learn. Res.*, **13**, 1967–1971.

Keilwagen,J. and Grau,J. (2015) Varying levels of complexity in transcription factor binding motifs. *Nucleic Acids Res.*, **43**, e119.

Mathelier,A. *et al*. (2016) DNA shape features improve transcription factor binding site predictions in vivo. *Cell Syst*., **3**, 278–286.

Nielsen,S. (2000) The stochastic EM algorithm: Estimation and asymptotic results. *Bernoulli*, **6**, 457–489.

Sandelin,A. *et al*. (2004) JASPAR: an open-access database for eukaryotic transcription factor binding profiles. *Nucleic Acids Res*., **32**, D91–D94.

Schneider,T. and Stephens,R. (1990) Sequence logos: A new way to display consensus sequences. *Nucleic Acids Res*., **18**, 6097–6100.

Schwarz,G.E. (1978) Estimating the dimension of a model. *Ann. Stat*., **2**, 461–464.

Siebert,M. and Söding,J. (2016) Bayesian Markov models consistently outperform PWMs at predicting motifs in nucleotide sequences. *Nucleic Acids Res*., **44**, 6055–6069.

Stormo,G. *et al*. (1982) Characterization of translational initiation sites in *E.coli*. *Nucleic Acids Res*., **10**, 2971–2996.

Zhao,Y. *et al*. (2012) Improved models for transcription factor binding site identification using nonindependent interactions. *Genetics*, **191**, 781–790.